

# Package: wISAM (via r-universe)

August 23, 2024

**Type** Package

**Title** Weighted Inbred Strain Association Mapping

**Version** 0.2.8

**Author** Robert W. Cort <rcorty@gmail.com>

**Maintainer** Robert W. Cort <rcorty@gmail.com>

**Description** In the course of a genome-wide association study, the situation often arises that some phenotypes are known with greater precision than others. It could be that some individuals are known to harbor more micro-environmental variance than others. In the case of inbred strains of model organisms, it could be the case that more organisms were observed from some strains than others, so the strains with more organisms have better-estimated means. Package 'wISAM' handles this situation by allowing for weighting of each observation according to residual variance. Specifically, the 'weight' parameter to the function `conduct_scan()` takes the precision of each observation (one over the variance).

**License** GPL-3

**Encoding** UTF-8

**Depends** R (>= 3.0.0)

**LazyData** true

**Suggests** testthat

**RoxygenNote** 6.0.1

**Imports** methods, MASS

**Repository** <https://rcorty.r-universe.dev>

**RemoteUrl** <https://github.com/rcorty/wisam>

**RemoteRef** HEAD

**RemoteSha** 9d90e9d810aecbf626b3cf20e84f8efe85450abe

## Contents

check_eigen_decomposition . . . . .	2
covariate_mat . . . . .	3
GenomeScan-class . . . . .	3
GenomeScan_calc_multiplier_eigen . . . . .	4
GenomeScan_conduct_scan . . . . .	4
GenomeScan_fit_locus . . . . .	5
GenomeScan_fit_locus_given_h2 . . . . .	5
GenomeScan_initialize . . . . .	5
GenomeScan_prep_scan . . . . .	6
kinship_mat . . . . .	6
locus_list . . . . .	7
phenotype . . . . .	7
se_mean_per_strain . . . . .	7
<b>Index</b>	<b>8</b>

---

check\_eigen\_decomposition  
*check\_eigen\_decomposition*

---

### Description

Grabbed from MASS. Useful to sparsify matrices when some eigenvalues are essentially zero.

### Usage

```
check_eigen_decomposition(e, tol = 1e-06)
```

### Arguments

e	the eigen decomposition to check
tol	the threshold below which a number is said to be effectively zero, defaults to 1e-6

### Value

The eigen values with any values with absolute value less than tol zeroed.

---

covariate_mat	<i>Example covariate matrix</i>
---------------	---------------------------------

---

**Description**

Example covariate matrix

**Usage**

covariate\_mat

**Format**

A matrix with 200 rows and 4 variables

---

GenomeScan-class	<i>GenomeScan</i>
------------------	-------------------

---

**Description**

A Reference Class implementing a Genome Scan

**Fields**

Options these are options

Data Things the user inputs. They have interpretable meaning and define the GenomeScan. Currently: y, X, G, K, weights (inverse variances), and variances.

Intermediates\_per\_scan Things the GenomeScan will compute once per scan. They are mathematical tools that can't really be interpreted. Currently: L, eigen\_L, and LL\_null.

Intermediates\_per\_locus Things the GenomeScan will compute once per locus. They are mathematical tools that can't really be interpreted. Currently: XG

Intermediates\_per\_fit Things the GenomeScan will compute many times per locus (once per trial fit on that locus). These are interpretable but rapidly changing and not guaranteed to be finalized or optimal. Currently: M, LDV, and h2

Results The results of the GenomeScan. Currently: The h2 that maximizes the LL at each locus and the LR as compared with the no-locus (null) model.

---

GenomeScan\_calc\_multiplier\_eigen  
*calc\_multiplier\_eigen*

---

**Description**

Compute a multiplier (aka rotation) matrix. Details in in h2lmm\_math\_RWC.Rmd.

**Value**

an object of class GenomeScan

---

GenomeScan\_conduct\_scan  
*conduct\_scan*

---

**Description**

Conducts the GenomeScan.

**Value**

an object of class GenomeScan

**Note**

TODO: allow user to specify subset of chromosomes or loci

**Examples**

```
library(wISAM)

wgs <- GenomeScan$new(y = phenotype,
                     X = covariate_mat,
                     G = locus_list,
                     K = kinship_mat,
                     w = 1/se_mean_per_strain)

result <- wgs$prep_scan()$conduct_scan()
```

---

GenomeScan\_fit\_locus *fit\_locus*

---

**Description**

Fit one locus of a GenomeScan. Should not typically be called by a user.

**Value**

an object of class GenomeScan

---

GenomeScan\_fit\_locus\_given\_h2  
*fit\_locus\_given\_h2*

---

**Description**

Fit one locus at a specified value of h2. Should not typically be called by a user.

**Value**

an object of class GenomeScan

---

GenomeScan\_initialize *initialize*

---

**Description**

Initialize a GenomeScan

**Arguments**

y	vector of length n - the phenotype of each of n genomes (individuals or strains)
X	matrix of dimension n-by-c - the covariate value of each individual for c covariates
G	a list where each element is of length n - the genotype of each individual at p loci
K	matrix of dimension n-by-n - the covariance of the phenotype
w	matrix of dimension n-by-n - the inverse variance of the phenotype

**Value**

an object of class GenomeScan

**Examples**

```
library(wISAM)

wgs <- GenomeScan$new(y = phenotype,
                      X = covariate_mat,
                      G = locus_list,
                      K = kinship_mat,
                      w = 1/se_mean_per_strain)
```

---

GenomeScan\_prep\_scan    *prep\_scan*

---

**Description**

Prepare a GenomeScan for running. Does all the computations that need to be done exactly once per genome scan.

**Value**

an object of class GenomeScan

**Examples**

```
library(wISAM)

wgs <- GenomeScan$new(y = phenotype,
                      X = covariate_mat,
                      G = locus_list,
                      K = kinship_mat,
                      w = 1/se_mean_per_strain)

result <- wgs$prep_scan()
```

---

kinship\_mat                    *Example kinship matrix*

---

**Description**

Example kinship matrix

**Usage**

```
kinship_mat
```

**Format**

A matrix with 200 rows and 200 columns

---

locus_list	<i>Example locus list</i>
------------	---------------------------

---

**Description**

Example locus list

**Usage**

locus\_list

**Format**

A list of matrices, where each matrix has 200 rows and 1 column

---

phenotype	<i>Example phenotype</i>
-----------	--------------------------

---

**Description**

Example phenotype

**Usage**

phenotype

**Format**

A vector of length 200

---

se_mean_per_strain	<i>Example standard error of the mean per strain</i>
--------------------	--

---

**Description**

Example standard error of the mean per strain

**Usage**

se\_mean\_per\_strain

**Format**

A vector of length 200

# Index

## \* data

- covariate\_mat, 3
- kinship\_mat, 6
- locus\_list, 7
- phenotype, 7
- se\_mean\_per\_strain, 7

- check\_eigen\_decomposition, 2
- covariate\_mat, 3

- GenomeScan (GenomeScan-class), 3
- GenomeScan-class, 3
- GenomeScan\_calc\_multiplier\_eigen, 4
- GenomeScan\_conduct\_scan, 4
- GenomeScan\_fit\_locus, 5
- GenomeScan\_fit\_locus\_given\_h2, 5
- GenomeScan\_initialize, 5
- GenomeScan\_prep\_scan, 6

- kinship\_mat, 6

- locus\_list, 7

- phenotype, 7

- se\_mean\_per\_strain, 7